



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
-----------------	-------------	----------------------	---------------------	------------------

09/921,543

08/02/2001

Charles L. Vigue

NAIIP275/01.014.01

6834

28875

7590

10/09/2009

Zilka-Kotab, PC

P.O. BOX 721120

SAN JOSE, CA 95172-1120

EXAMINER

HENNING, MATTHEW T

ART UNIT

PAPER NUMBER

2431

NOTIFICATION DATE

DELIVERY MODE

10/09/2009

ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

anita@zilkakotab.com

erica@zilkakotab.com

dottie@zilkakotab.com

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES

Ex parte CHARLES L. VIGUE, MARTIN FALLENSTEDT,
DANIEL MELCHIONE, and VICTOR KOUZNETSOV

Appeal 2008-005225
Application 09/921,543
Technology Center 2400

Decided: October 7, 2009

Before JEAN R. HOMERE, ST. JOHN COURTENAY III, and
STEPHEN C. SIU, *Administrative Patent Judges*.

SIU, *Administrative Patent Judge*.

DECISION ON APPEAL
STATEMENT OF THE CASE

This is a decision on appeal under 35 U.S.C. § 134(a) from the Examiner's rejections of claims 1-11 and 13-25. We have jurisdiction under 35 U.S.C. § 6(b).

We affirm.

Invention

The invention relates to a system and method for secure and verified sharing of resources in a peer-to-peer network environment to facilitate efficient use of bandwidth. (Spec. 1, ll. 21-23).

Independent claim 1 is illustrative:

1. A method for securely sharing resources over a peer-to-peer network, comprising:

broadcasting a single request to a plurality of peers by a requesting peer for a resource over the peer-to-peer network wherein the request contains an identification of the resource and the resource identification contains a resource version identifier;

receiving a response from a responding peer on the peer-to-peer network indicating that the responding peer has the requested resource;

retrieving the requested resource from the responding peer; and

verifying the retrieved resource by ensuring the retrieved resource contains the version identifier embedded therein.

References

The Examiner relies upon the following references as evidence in support of the rejections:

Shostack	US 6,298,445 B1	Oct. 2, 2001
Peng	US 6,317,754 B1	Nov. 13, 2001
Delaney	US 6,374,289 B2	Apr. 16, 2002
Radatti	US 2002/0170052 A1	Nov. 14, 2002

VeriSign, Verisign gets US approval for 128-bit key certificates export (July 14, 1997) (“Verisign”).

Rejections

Claims 1, 4-7, 15, and 18-25 are rejected under 35 U.S.C. § 103(a) as being unpatentable over Peng and Delaney.

Claims 2 and 16 are rejected under 35 U.S.C. § 103(a) as being unpatentable over Peng, Delaney, and Shostack.

Claims 3 and 17 are rejected under 35 U.S.C. § 103(a) as being unpatentable over Peng, Delaney, Shostack, and Verisign.

Claims 8 and 11 are rejected under 35 U.S.C. § 103(a) as being unpatentable over Radatti and Delaney.

Claims 9 and 13 are rejected under 35 U.S.C. § 103(a) as being unpatentable over Radatti, Delaney, and Shostack.

Claims 10 and 14 are rejected under 35 U.S.C. § 103(a) as being unpatentable over Radatti, Delaney, Shostack, and Verisign.

ISSUE 1

Appellants assert “that it would not have been obvious to combine the teachings of the Delaney and Peng references” (App. Br. 11).

Issue: Did Appellants demonstrate that the Examiner erred in finding that it would have been obvious to one of ordinary skill in the art to combine the Delaney and Peng references?

ISSUE 2

Appellants argue that, where relied upon, “Peng merely teaches verifying that a received object ‘has a version identifier or time stamp older than or equal to the version vector of the corresponding object in the first [receiving] server’” (App. Br. 12 (alteration in original)).

Issue: Did Appellants demonstrate that the Examiner erred in finding that Peng teaches or suggests (1) a request that contains an identification of a resource and the resource identification contains a resource version identifier and (2) verifying a retrieved resource by ensuring that the retrieved resource contains the version identifier embedded therein?

ISSUE 3

Appellants argue that, where relied upon, Peng “merely relates to synchronizing two servers, and not ‘determin[ing] which resources are to be requested over the peer-to-peer network’” (App. Br. 14 (alteration in original)).

Issue: Did Appellants demonstrate that the Examiner erred in finding that Peng and Delaney teach or suggest comparing a listing of resources with resources installed at the requesting peer to determine which resources are to be requested over a peer-to-peer network?

ISSUE 4

Appellants argue that, where cited, Radatti only teaches “determining if a ‘server target file hash does not match the client entry in the update_index file’” (App. Br. 17).

Issue: Did Appellants demonstrate that the Examiner erred in finding that Radatti teaches verifying retrieved resources by ensuring that each retrieved resource contains a requested version identifier embedded therein?

FINDINGS OF FACT

The following Findings of Fact (FF) are shown by a preponderance of the evidence.

1. Peng teaches a system “for synchronizing servers which accommodates wide area mobile computing” (Abstract). In this system, “synchronization from a mobile computer can be done whether in client/server mode, or peer-to-peer” (*id.*).
2. Delaney teaches a system “for enabling data package distribution to be performed by a plurality of peer clients” (Abstract). In this system, “the peer client downloads the data package or data packages” (col. 8, ll. 6-7).
3. Peng teaches that “[t]he first server sends its summarizing version vector to the second server” (col. 5, ll. 41-42). “[S]ummarizing version vector . . . means a vector having fields which summarize the state of the object container at a server” (col. 3, ll. 15-17).
4. Peng teaches that “[u]pon receiving each object and update from the second server . . . [i]f the received object or update has a version vector or time stamp older than or equal to

the version vector of the corresponding object in the first server, this object or update will be thrown away” (col. 6, ll. 51-57).

5. Peng teaches that “[u]pon receiving the summarizing version vector and identifiers from the second server, the first server figures out all of the identifiers of objects which need to be received as whole objects and sent to the second server” (col. 5, ll. 48-52). This includes the sub-step of finding “all objects which exist in the second server but do not exist in the first server and put[ting] their identifiers into the list of objects needed to be received from the second server” (col. 5, ll. 53-56). It also includes the sub-step of sending “all the identifiers obtained . . . to the second server” (col. 6, ll. 5-6).

6. Radatti teaches that “[t]he local or client machine also contains a copy of the update file and the hash of the update file” (¶ [0012]). “The hash of the update file is transmitted from the distribution media to the local machine and compared to the local version of the hash of the update file. If the versions are identical, no update is necessary” (¶ [0013]).

7. Radatti gives an example of a request for an update hash, http://update.cybersoft.com/verygoodprogram/1.0/update_hash (¶ [0049]). The file `update_index` contains the entries “FILE01:abc1:1.0:0:0:XXX1” (¶ [0080]) and “FILE02:xyz2:1.0:0:0:XXX2” (¶ [0081]). “The local `update_hash` file contains, in this embodiment, the hash for

these entries” (§ [0082]). “If another version of software product [*sic*] is available, the hash is different” (§ [0083]).

8. Radatti teaches that

Any embodiments which obtain or retain file name and version information or any other identifying information from the client machine may also be used to insure file integrity through hashing the data information and comparing the hash to appropriate server software product information. If the hashes do not match, automatic or manual processes may be used to notify the user, correct any problems, etc.

For example . . . if the server target file hash does not match the client entry in the update_index file, update_manager will search for references to other modules. [For example, when update_manager encounters “OTHER FROM 2.8 TO 2.9 NAME:FILE61”] it will download the FILE61 module and check the target hash in that module against the target hash in the client update_index file. If that target hash matches, update_manager will proceed with the update. . . . This recursive module mechanism of this embodiment provides the client with the information that previous version or versions exist.

(§§ [0093-94]).

PRINCIPLES OF LAW

Claim interpretation

“In the patentability context, claims are to be given their broadest reasonable interpretations. . . . [L]imitations are not to be read into the

claims from the specification.” *In re Van Geuns*, 988 F.2d 1181, 1184 (Fed. Cir. 1993) (citations omitted).

Obviousness

The question of obviousness is resolved on the basis of underlying factual determinations including (1) the scope and content of the prior art, (2) any differences between the claimed subject matter and the prior art, and (3) the level of skill in the art. *Graham v. John Deere Co.*, 383 U.S. 1, 17-18 (1966). One cannot show nonobviousness by attacking references individually where the rejections are based on combinations of references. *In re Merck & Co., Inc.*, 800 F.2d 1091, 1097 (Fed. Cir. 1986).

“The combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results,” *KSR Int’l Co. v. Teleflex, Inc.*, 550 U.S. 398, 416 (2007), especially if the combination would not be “uniquely challenging or difficult for one of ordinary skill in the art,” *Leapfrog Enters., Inc. v. Fisher-Price, Inc.*, 485 F.3d 1157, 1162 (Fed. Cir. 2007) (citing *KSR*, 550 U.S. at 418).

“A reference may be said to teach away when a person of ordinary skill, upon reading the reference, would be discouraged from following the path set out in the reference, or would be led in a direction divergent from the path that was taken by the applicant.” *In re Gurley*, 27 F.3d 551, 553 (Fed. Cir. 1994).

ANALYSIS

Issue 1

We agree with the Examiner that it would have been obvious to one of ordinary skill in the art to have combined the Peng and Delaney references. Appellants admit that “Peng relates to synchronizing servers, while Delaney relates to distributing data packages among peer clients” (App. Br. 11; FF 1, 2). Appellants, however erroneously conclude that Peng and Delaney are non-analogous (App. Br. 11).

Peng “allows for a pair of servers to exchange data such that each resultant server contains the same data” (App. Br. 11). Delaney “allows for data packages to be requested from peers to other peers where such other peers may respond to the request and only the requested data packages may be downloaded” (*id.*). Based on Appellants’ admissions, the types of problems solved by Delaney (one-way distribution of data) are a subset of the types of problems solved by Peng (two-way distribution of data). Moreover, Peng is adaptable to leverage peer-to-peer technologies (FF 1). An artisan of ordinary skill would find Peng and Delaney to be analogous. Thus, their teachings are combinable.

Appellants further argue that Delaney teaches away from claims 7 and 21 (App. Br. 14-15). We do not agree. Appellants note that Delaney teaches that multiple data packages can optionally be requested in one message (App. Br. 15). Delaney, however, also teaches that peer clients can download a single data package (FF 2). An artisan of ordinary skill would know that an explicitly optional multiple data package request teaching need

not be used. An artisan of ordinary skill would find it obvious that multiple data packages could be downloaded as a series of single data package downloads. Therefore an artisan of ordinary skill would not be discouraged from or led away from requesting each resource in a separate transaction. *See In re Gurley*, 27 F.3d at 553.

For at least these reasons, we find that Appellants have not sustained the requisite burden on appeal in providing arguments or evidence persuasive of error in the Examiner's 35 U.S.C. § 103(a) rejections of claims 1-7 and 15-25 with respect to this issue.

Issue 2

We agree with the Examiner that Peng teaches (1) a request that contains an identification of a resource and the resource identification contains a resource version identifier and (2) verifying a retrieved resource by ensuring that the retrieved resource contains the version identifier embedded therein.

Peng teaches that a first server sends a summarizing version vector to a second server (FF 3). The summarizing version vector summarizes the state of the object container at the first server (FF 3). Thus, the summarizing version vector is a request containing an identification of a resource that contains a resource version identifier. Peng also teaches that the first server receives each object and update from the second server (FF 4). Peng then teaches that the first server checks whether the received object or update has a version vector or time stamp older than or equal to the version vector of

the corresponding object in the first server (FF 4). Therefore, Peng teaches (1) a request that contains an identification of a resource and the resource identification contains a resource version identifier (the summarizing vector) and (2) verifying (checking whether) a retrieved resource (received objects or updates) by ensuring that the retrieved resource contains the version identifier embedded therein (a version vector or time stamp equal to that of the corresponding local object).

Appellants argue that Peng does not teach “ensuring the retrieved resource has the originally requested version identifier” (App. Br. 13). This is not persuasive. The summarizing version vector identifies the local object versions (FF 3). Peng ensures that the resource as the original requested version identifier by comparing received resource version vectors and time stamps with those of local objects (FF 4).

Appellants also protest that Peng does not teach the claimed invention because “received objects with older version vectors are thrown away” (App. Br. 13). This is not persuasive because the claims are silent regarding steps subsequent to verification. Limitations are not to be read into the claims from the specification. *See In re Van Geuns*, 988 F.2d at 1184.

We are also not persuaded by Appellants’ arguments that Peng teaches comparison of version vectors two times (App. Br. 13). Appellants’ claims are open-ended. The claims do not preclude multiple comparisons.

For at least these reasons, we find that Appellants have not sustained the requisite burden on appeal in providing arguments or evidence

persuasive of error in the Examiner's 35 U.S.C. § 103(a) rejections of claims 1-7 and 15-25 with respect to this issue.

Issue 3

We agree with the Examiner that Peng and Delaney teach or suggest comparing a listing of resources with resources installed at the requesting peer to determine which resources are to be requested over a peer-to-peer network.

Peng teaches that a first server receives identifiers (a listing of resources) from a second server (FF 5). The first server then figures out all of the object identifiers that need to be requested (FF 5). This is done by finding objects which exist in the second server but not the first server (comparing a listing of resources with locally installed resources) (FF 5). Identifiers for objects that do not exist in the first server (resources to be requested) are then sent to the second server (FF 5).

Appellants argue that these teachings do not extend to peer-to-peer networks (App. Br. 14). They emphasize the “second server” language of Peng (Reply Br. 8). But Appellants cannot show nonobviousness by attacking Peng alone. The Examiner's rejection is based on the combination of Peng and Delaney. *See In re Merck & Co., Inc.*, 800 F.2d at 1097.

For at least these reasons, we find that Appellants have not sustained the requisite burden on appeal in providing arguments or evidence persuasive of error in the Examiner's 35 U.S.C. § 103(a) rejections of claims 6, 7, 20, and 21 with respect to this issue.

Issue 4

We agree with the Examiner that Radatti teaches verifying retrieved resources by ensuring that each retrieved resource contains a requested version identifier embedded therein.

Radatti teaches that a client machine contains a copy of an update file and a hash of the update file (FF 6). The update file contains a version identifier (FF 7). The client system obtains a remote update file hash (FF 6). The system checks whether the remote hash is the same as the local hash (FF 6).

An example in Radatti shows a request that includes a version identifier (FF 7). The update file contains the same version identifier (FF 7). The hash changes if the version identifier changes (FF 7). Because of this dependency, the hash value embeds the version identifier. Thus, Radatti teaches a system that verifies (checks) whether a retrieved resource (the remote hash) contains a requested version identifier (from the request) embedded therein (the hash changes if the version identifier changes). Furthermore, Radatti teaches this verification can be done for multiple retrieved resources (target hash values) (FF 8).

For at least these reasons, we find that Appellants have not sustained the requisite burden on appeal in providing arguments or evidence persuasive of error in the Examiner's 35 U.S.C. § 103(a) rejections of claims 8-11, 13, and 14 with respect to this issue.

CONCLUSIONS OF LAW

Based on the findings of facts and analysis above, we conclude that Appellants have not demonstrated:

1. that the Examiner erred in finding that it would have been obvious to one of ordinary skill in the art to combine the Delaney and Peng references (Issue 1);

2. that the Examiner erred in finding that Peng teaches or suggests (1) a request that contains an identification of a resource and the resource identification contains a resource version identifier and (2) verifying a retrieved resource by ensuring that the retrieved resource contains the version identifier embedded therein (Issue 2);

3. that the Examiner erred in finding that Peng and Delaney teach or suggest comparing a listing of resources with resources installed at the requesting peer to determine which resources are to be requested over a peer-to-peer network (Issue 3); and

4. that the Examiner erred in finding that Radatti teaches verifying retrieved resources by ensuring that each retrieved resource contains a requested version identifier embedded therein (Issue 4).

DECISION

We affirm the Examiner's decisions rejecting claims 1-11 and 13-25 under 35 U.S.C. § 103(a).

No time period for taking any subsequent action in connection with this appeal may be extended under 37 C.F.R. § 1.136(a)(1)(iv).

Appeal 2008-005225
Application 09/921,543

AFFIRMED

msc

Zilka-Kotab, PC
P.O. BOX 721120
SAN JOSE CA 95172-1120